

การจัดลำดับการผลิตแบบหลายวัตถุประสงค์ บนสายการประกอบผลิตภัณฑ์ผสมแบบสองด้านภายใต้ผลกระทบจากการเรียนรู้

วันวิสา นฤมิตวงศ์¹ และ ปารเมศ ชุตินา^{2*}

บทคัดย่อ

การจัดลำดับการผลิตมีความสำคัญอย่างยิ่งในการช่วยแก้ปัญหาสายการประกอบผลิตภัณฑ์ผสมแบบสองด้าน เพื่อให้การผลิตมีประสิทธิภาพสูงสุด ซึ่งปัญหาการจัดลำดับการผลิตนั้นต้องพิจารณาหลายปัจจัย ทั้งฟังก์ชันวัตถุประสงค์ที่มีหลายตัว และผลกระทบจากการเรียนรู้ จึงทำให้ปัญหามีความยุ่งยากซับซ้อนมากยิ่งขึ้น (ปัญหาแบบ Non-deterministic Polynomial-Hard : NP-Hard) งานวิจัยนี้ได้นำอัลกอริทึมที่มีชื่อว่า วิธีการหาค่าเหมาะสมแบบการกระจายของสิ่งมีชีวิตตามภูมิศาสตร์ (Biogeography-based Optimization: BBO) เข้ามาประยุกต์ใช้ในการแก้ปัญหาการจัดลำดับการผลิต โดยจะพิจารณาฟังก์ชันวัตถุประสงค์ 3 ฟังก์ชันวัตถุประสงค์ คือ ความแปรผันของการผลิตน้อยที่สุด ปริมาณงานที่ไม่เสร็จน้อยที่สุด และ

เวลาการปรับตั้งเครื่องน้อยที่สุด แล้วทำการเปรียบเทียบประสิทธิภาพกับอัลกอริทึมที่ได้รับการยอมรับว่าสามารถแก้ปัญหาประเภทนี้ได้อย่างมีประสิทธิภาพ ได้แก่ NSGA-II และ DPSO จากการเปรียบเทียบประสิทธิภาพของผลการวิจัย พบว่า BBO มีประสิทธิภาพด้านการลู่เข้าสู่กลุ่มคำตอบ 98.67 เปอร์เซ็นต์และด้านอัตราส่วนของจำนวนกลุ่มคำตอบที่ค้นพบเทียบกับกลุ่มคำตอบที่แท้จริง 76.13 เปอร์เซ็นต์ ซึ่งดีกว่า NSGAII และ DPSO

คำสำคัญ: วิธีการหาค่าเหมาะสมแบบการกระจายของสิ่งมีชีวิตตามภูมิศาสตร์การจัดลำดับการผลิตสายการประกอบผลิตภัณฑ์ผสมแบบสองด้าน ฟังก์ชันหลายวัตถุประสงค์ ผลกระทบจากการเรียนรู้

¹ นิสิต ภาควิชาวิศวกรรมอุตสาหการ คณะวิศวกรรมศาสตร์ จุฬาลงกรณ์มหาวิทยาลัย

² รองศาสตราจารย์ ภาควิชาวิศวกรรมอุตสาหการ คณะวิศวกรรมศาสตร์ จุฬาลงกรณ์มหาวิทยาลัย

* ผู้นิพนธ์ประสานงาน โทรศัพท์ 0-2218-6847 อีเมล: cparames@chula.ac.th



Multi-Objective Sequencing Problem on Mixed-Model Two-sided Assembly Lines under Learning Effect

Wanwisa Naruemitwong¹ and Parames Chutima^{2*}

Abstract

Sequencing is an important method for solving the mixed-model two-sided assembly lines problem to reach maximum production efficiency. Many factors such as multiple objectives and learning effect have to be considered in solving the sequencing problem. These make the problem more complicated as known as “NP-Hard problem”. In this research, a Biogeography-based Optimization (BBO) algorithm is adopted for solving the sequencing problem to minimize variance of production rates, utility work, and setup time. The results is compared

with well-known algorithms such as Non-dominated Sorting Genetic Algorithms (NSGA-II) and Discrete Particle Swarm Optimization (DPSO). The experiments show that BBO has performed of convergence 98.67% and ratio 76.13%, consequently, which are better than NSGA-II and DPSO.

Keywords: Biogeography-based Optimization, Sequencing, Mixed-model two-sided Assembly Line, Multi-objective Functions, Learning Effect

¹ Student, Department of Industrial Engineering Faculty of Engineering, Chulalongkorn University.

² Associate Professor, Department of Industrial Engineering Faculty of Engineering, Chulalongkorn University.

* Corresponding Author, Tel. 0-2218-6847, E-mail: cparames@chula.ac.th

1. บทนำ

ปัจจุบันนี้โรงงานอุตสาหกรรมได้มีการออกแบบสายการประกอบให้เป็นแบบผลิตภัณฑ์ผสม ซึ่งเป็นสายการประกอบที่สามารถผลิตได้หลายผลิตภัณฑ์ ผลิตภัณฑ์ที่มีความแตกต่างกันจะเข้าสู่สายการประกอบพร้อมๆ กัน และต่อเนื่องกัน ปัญหาในการนำสายการประกอบแบบผลิตภัณฑ์ผสมมาใช้ คือ การจัดลำดับการผลิตเพื่อให้การผลิตมีประสิทธิภาพตามวัตถุประสงค์ที่โรงงานต้องการ วัตถุประสงค์ที่ใช้กันทั่วไปมีหลายวัตถุประสงค์ เช่น ปริมาณงานที่ไม่เสร็จในการผลิตน้อยที่สุด อัตราความผันแปรของการผลิตน้อยที่สุด ความสม่ำเสมอของการใช้ชิ้นส่วนประกอบ ต้นทุน/เวลาในการปรับตั้งเครื่องน้อยที่สุดความสม่ำเสมอของปริมาณงาน ความเสี่ยงในการหยุดของสายการประกอบ มีค่าน้อยที่สุด ฯลฯ [1] การแก้ปัญหาการจัดลำดับการผลิตในภาคปฏิบัติจึงต้องพิจารณาหลายวัตถุประสงค์ไปพร้อมๆ กัน ทำให้ปัญหามีความซับซ้อนยุ่งยากมากยิ่งขึ้น เวลาที่ใช้ในการคำนวณยาวนานมากขึ้น จำนวนคำตอบที่ได้มากมาย จึงมีการศึกษาวิจัยวิธีการเพื่อที่จะหาคำตอบโดยการนำวิธีฮิวริสติกเข้ามาช่วยแก้ปัญหา เช่น วิธีเจเนติกอัลกอริทึม (Genetic Algorithms: GAs) ซึ่งเป็นวิธีที่นิยมนำมาใช้แก้ปัญหาในการจัดลำดับการผลิต เมื่อเปรียบเทียบกับอัลกอริทึมอื่น GA สามารถลดเวลาในการคำนวณและให้คำตอบที่ใกล้เคียงกับคำตอบที่ดีที่สุด Hyun และคณะ [2] ได้พัฒนาการค้นหาคำตอบในการจัดลำดับการผลิตที่มีหลายวัตถุประสงค์ มีชื่อว่า Pareto Stratum-Niche Cubic (PS-NC) วัตถุประสงค์ที่ใช้มี 3 วัตถุประสงค์ คือ ผลรวมของปริมาณงานที่ไม่เสร็จมีค่าน้อยที่สุด ความสม่ำเสมอของอัตราการใช้ชิ้นส่วนประกอบ และผลรวมของค่าใช้จ่ายในการปรับตั้งเครื่องมีค่าน้อยที่สุด McMullen และคณะ [3] ได้มีการพิจารณาวัตถุประสงค์ 2 วัตถุประสงค์ คือ จำนวนครั้งในการปรับตั้งมีค่าน้อยที่สุด และความสม่ำเสมอของอัตราการใช้ชิ้นส่วนประกอบ Mansouri [4] ได้มีการแก้ปัญหาการจัดลำดับที่มีหลายวัตถุประสงค์ (Multi-Objective Genetic Algorithm: MOGA) ซึ่งในงานวิจัยสรุปว่า MOGA มีประสิทธิภาพในการนำมาใช้แก้ปัญหาในการจัดลำดับการผลิต Vahed และคณะ [5] ได้พัฒนา

เจเนติกอัลกอริทึมที่มีชื่อว่า Multi-Objective Scatter Search (MOSS) โดยใช้วัตถุประสงค์เหมือนกับงานวิจัยของ Hyun และคณะ [2] พบว่าเมื่อเปรียบเทียบกับ Non-dominated Sorting Genetic Algorithms: NSGA-II และ PS-NC แล้ว ประสิทธิภาพของ MOSS ดีกว่า แต่ MOSS ใช้เวลาในการคำนวณมากกว่า เพราะฉะนั้นในการพัฒนาอัลกอริทึมจึงควรพัฒนาประสิทธิภาพในการหาคำตอบควบคู่ไปกับเวลาที่ใช้หาคำตอบ

Simon [6] ได้มีการศึกษาและพัฒนาอัลกอริทึมใหม่ที่มีชื่อว่า วิธีการหาค่าเหมาะสมแบบการกระจายของสิ่งมีชีวิตตามภูมิศาสตร์ (Biogeography Based Optimization: BBO) เป็นการนำแนวคิดการเคลื่อนย้ายประชากรจากแหล่งหนึ่งไปอีกแหล่งหนึ่งมาใช้ โดยวิธีการหาค่าเหมาะสมแบบการกระจายของสิ่งมีชีวิตตามภูมิศาสตร์นี้ได้มีการเปรียบเทียบคำตอบที่ได้จากปัญหาในภาคปฏิบัติ คำตอบที่ได้จึงมีประสิทธิภาพดีกว่าอัลกอริทึมหลายตัว Abell and Du [7] ได้นำเอา BBO มาใช้แก้ปัญหาที่มีความซับซ้อนหลายวัตถุประสงค์ และมีหลายข้อจำกัดของปัญหา Complex System Families และ Mo and Xu [8] ได้นำ BBO มาประยุกต์ใช้ในการแก้ปัญหา Traveling Salesman จากงานวิจัยที่กล่าวมา พบว่า BBO มีประสิทธิภาพในการแก้ปัญหา มากกว่าเมื่อเปรียบเทียบกับ GA คำตอบที่ได้จาก BBO มีโอกาสที่จะใกล้เคียงกับคำตอบที่ดีที่สุด เพราะคำตอบที่ออกมา นั้นเกิดมาจากคำตอบของหลายคำตอบ ซึ่งจะไม่เหมือน GA ที่คำตอบเกิดมาจากคำตอบของพ่อแม่เพียงคู่เดียว

ในการหาคำตอบผลกระทบจากการเรียนรู้มีผลต่อเวลาในการดำเนินการ ซึ่งจะทำให้เวลาในการผลิตลดลง [9] จึงมีผลต่อค่าวัตถุประสงค์ที่ทำการวิจัย ดังนั้นผู้วิจัยจึงนำ BBO มาประยุกต์ใช้ในการจัดลำดับการผลิตที่มีหลายวัตถุประสงค์บนสายการประกอบผลิตภัณฑ์ผสมแบบสองด้านภายใต้ผลกระทบจากการเรียนรู้และทำการเปรียบเทียบประสิทธิภาพในการค้นหาคำตอบของ BBO กับอัลกอริทึมที่ได้รับการยอมรับว่าสามารถแก้ ปัญหาประเภทนี้ได้ ได้แก่ NSGAII และ DPSO [7], [8], [10]

2. สายการประกอบแบบสองด้าน

สายการประกอบแบบสองด้าน (Two-Sided Assembly Line) เป็นสายการประกอบที่มีลักษณะเป็นเส้นตรง แต่จะมีสถานีงานอยู่ทั้งด้านซ้าย (L) และด้านขวา (R) ดังรูปที่ 1 โดยสายการประกอบนี้เหมาะสมกับผลิตภัณฑ์ที่มีขนาดใหญ่ และมีปริมาณการผลิตสูง เช่น รถยนต์ รถบรรทุก สายการประกอบนี้ทำให้ความยาวของสายการประกอบสั้นลงเป็นผลให้ช่วยลดระยะเวลาในการเคลื่อนที่ของพนักงาน ลดเวลาในการปรับตั้งเครื่องจักรและลดเวลาในการผลิตให้สั้นลง และเครื่องมืออุปกรณ์ในสถานีงานประเภทนี้สามารถใช้ร่วมกันได้จึงทำให้ต้นทุนในการผลิตลดลง [11]

ในการจัดสรรชิ้นงานลงในสถานีงานบนสายการประกอบแบบสองด้าน มีข้อจำกัดในการจัดสรรชิ้นงาน 2 แบบ คือ ชิ้นงานที่สามารถจัดสรรลงในสถานีงานได้เพียงด้านเดียวเท่านั้น คือด้านซ้าย (L-type) หรือด้านขวา (R-type) ของสายการประกอบ และชิ้นงานที่สามารถจัดสรรลงในสถานีงานด้านใดด้านหนึ่งของสายการประกอบก็ได้ (E-type) สถานีงานที่อยู่คู่กัน เรียกว่า Mated-Station ในรูปที่ 1 Mated-Station ที่ 1 คือ สถานีงานที่ 1 กับสถานีงานที่ 2 และ Mated-Station ที่ 2 คือ สถานีงานที่ 3 กับสถานีงานที่ 4 [12]

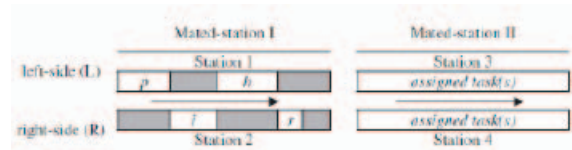
จากรูปที่ 2 ในสถานีงานที่ 1 ประกอบด้วยชิ้นงาน p และชิ้นงาน h ส่วนในสถานีงานที่ 2 ประกอบด้วยชิ้นงาน i และชิ้นงาน r ในการจัดสรรชิ้นงานจะมีเงื่อนไขลำดับงานก่อนหลัง เช่น ชิ้นงาน i จะต้องรอให้ชิ้นงาน p เสร็จก่อนจึงจะสามารถเริ่มดำเนินงานได้ และชิ้นงาน h จะเริ่มได้ก็ต่อเมื่อชิ้นงาน i เสร็จ ทำให้สายการประกอบนี้มีส่วนที่ร่นเวลาเกิดขึ้น ส่วนนี้จะเรียกว่า เวลาเดินเปล่า (Idle Time) ซึ่งทำให้สายการประกอบแบบสองด้านมีความแตกต่างจากสายการประกอบแบบเส้นตรง เวลาดำเนินงานในสถานีงานจะรวมเวลาเดินเปล่าที่เกิดขึ้นเข้าไปด้วย

3. ผลกระทบจากการเรียนรู้

Wright [13] พบว่าเวลาที่ใช้ในการผลิตผลิตภัณฑ์ชนิดเดียวกันที่มีการผลิตแบบต่อเนื่องกันนั้นมียาลลดลง



รูปที่ 1 สายการประกอบแบบสองด้าน [12]



รูปที่ 2 สายการประกอบแบบสองด้าน [11]

โดยได้ผลผลิตเพิ่มขึ้นแต่ค่าใช้จ่ายต่อหน่วยจะลดลง แสดงว่าผลกระทบจากการเรียนรู้ (Learning Effects) มีผลต่อเวลาในการผลิต โดยสามารถคำนวณได้จากสมการที่ 1

$$p_{[k]} = p_{[1]}k^a \quad (1)$$

โดย

$p_{[k]}$ คือเวลาการทำงานของผลิตภัณฑ์ชิ้นที่ k ของการผลิตผลิตภัณฑ์จำนวน k ชิ้น

$p_{[1]}$ คือเวลาทำงานของผลิตภัณฑ์ชิ้นแรก

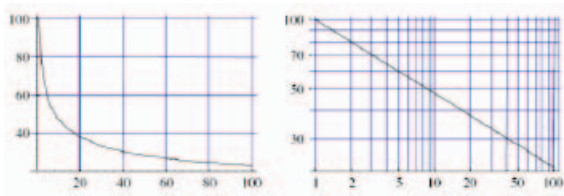
a คือค่า Learning Index ค่าของ a จะขึ้นอยู่กับค่า Learning Rate (LR)

$$a = \log LR / \log 2 = \log_2 LR \quad (2)$$

ดังนั้น เมื่อกำหนดให้ค่าการเรียนรู้เท่ากับ 80% $a = \log_2 0.8 = -0.322$ และกำหนดให้เวลาการทำงานของผลิตภัณฑ์ชิ้นแรกเท่ากับ 100 ดังนั้น

$$p_{[k]} = 100k^{-0.322} \quad (3)$$

จากสมการที่ 3 ถ้า k มีค่าเพิ่มขึ้นเรื่อยๆ จะทำให้เวลาการทำงานลดลง เมื่อนำจุดเวลามาพล็อตต่อกันจะเป็นเส้นกราฟดังแสดงในรูปที่ 3 กราฟเส้นโค้งการเรียนรู้ที่ค่า LR เท่ากับ 80% กราฟแรกจะเป็นกราฟปกติ กราฟที่สองเป็นกราฟ Double-Logarithmic โดยแกน y เป็นแกนของ



รูปที่ 3 กราฟเส้นโค้งการเรียนรู้ที่ค่า LR เท่ากับ 80% [14]

เวลาที่เริ่มต้นจาก 100 และแกน x เป็นแกนของจำนวนชิ้นของผลิตภัณฑ์ที่ผลิต

ในงานวิจัยนี้จะนำผลกระทบจากการเรียนรู้เข้ามาประยุกต์ใช้ในการจัดลำดับการผลิต เพื่อช่วยลดช่องว่างระหว่างเวลาในภาคทฤษฎีให้เข้าใกล้ภาคปฏิบัติมากขึ้น โดยใช้ผลกระทบจากการเรียนรู้แบบ Position-Based [14] ซึ่งสามารถคำนวณได้จากสมการ

$$p_{[im]} = p_{[im]} r^a \quad (4)$$

โดย

$p_{[im]}$ คือเวลาการดำเนินงานของชิ้นงานที่ i ของผลิตภัณฑ์ m

r คือจำนวนการทำซ้ำของชิ้นงานที่ i ของทุกผลิตภัณฑ์

a คือค่า Learning Index

ตัวอย่างการจัดลำดับการผลิตผลิตภัณฑ์ A, B และ C เวลาดำเนินงานของ A, B และ C ในชิ้นงานที่ 1 คือ 2, 3 และ 1 ตามลำดับ กำหนดให้ค่า LR เท่ากับ 80% และกำหนดให้มีสัดส่วนผลิตภัณฑ์ (MPS) = 1:2:3 แสดงว่าในการจัดลำดับการผลิตนี้จะมีผลิตภัณฑ์ A, B และ C จำนวน 1, 2 และ 3 ชิ้น ตามลำดับ โดยมีการจัดลำดับการผลิต (Model Sequence) เป็น ABCBCC โดยจะเริ่มจากการทำผลิตภัณฑ์ A ก่อน ต่อด้วยผลิตภัณฑ์ B และ C ไปเรื่อยๆ จนครบลำดับการผลิต

จากสมการที่ 2 ค่า a จะมีค่าเท่ากับ $a = \log_2 0.8 = -0.322$ และจะเริ่มพิจารณาเวลาดำเนินงานของลำดับการผลิตผลิตภัณฑ์ เช่น ผลิตภัณฑ์ A เป็นชิ้นงานแรกที่เริ่มทำเพราะฉะนั้น $r = 1$ จากสมการที่ 4 แทนค่าได้เป็น $p_{[1A]} = 2(1)^{-0.322} = 2$ เวลาดำเนินงาน

ของ A จึงเท่ากับ 2 ชิ้นงานที่ 1 ของผลิตภัณฑ์ B ถูกจัดในลำดับที่ 2 ดังนั้น $r = 2$ เวลาดำเนินงานจึงเท่ากับ $p_{[1B]} = 3(2)^{-0.322} = 2.3999$ จะเห็นได้ว่าเวลาดำเนินงานของ B มีค่าลดลง เนื่องจากเคยทำชิ้นงานที่ 1 จากผลิตภัณฑ์ A มาแล้ว

4. การหาค่าเหมาะสมที่สุดที่มีหลายวัตถุประสงค์

ในการพิจารณาแก้ปัญหาที่มีหลายวัตถุประสงค์ให้ มีประสิทธิภาพดีไปพร้อมกัน เป็นเรื่องที่แทบจะเป็นไปไม่ได้ เพราะจะไม่มีคำตอบที่ดีที่สุด [15], [16] จึงมีการนำเทคนิคในการค้นหาเซตคำตอบที่อยู่ในพื้นที่ของคำตอบที่เป็นไปได้ทั้งหมด (Feasible Region) มาแก้ปัญหา ในงานวิจัยนี้เป็นการหาค่าเหมาะสมที่สุดเพื่อหาค่าที่น้อยที่สุด ดังสมการที่ 5

$$\text{Minimize } \{f_1(\vec{x}), f_2(\vec{x}), \dots, f_k(\vec{x})\} \quad (5)$$

โดย

$\vec{x}$ คือเวกเตอร์ของตัวแปรตัดสินใจ

$f_i(\vec{x})$ คือฟังก์ชันวัตถุประสงค์ที่ i

โดยได้นำเทคนิคการกำหนดค่าความแข็งแรงด้วยวิธีเชิงกลุ่มที่ดีที่สุด (Pareto-based Approach) ที่มีการจัดอันดับคำตอบแบบพาเรโต (Pareto Ranking Approach) มาใช้ คำตอบที่ได้จะอยู่ในรูปของกลุ่มคำตอบที่ดีที่สุด (Pareto Optimal Set หรือ Pareto Optimal Front) ถ้าเวกเตอร์ของตัวแปรตัดสินใจ x ให้คำตอบที่ดีกว่าเวกเตอร์ของตัวแปรตัดสินใจ y แล้ว จะได้ว่า $f_i(x) \leq f_i(y)$ สำหรับทุกค่า $i \in \{1, 2, \dots, k\}$ และ $f_i(x) < f_i(y)$ มีอย่างน้อย 1 ค่าของ $i \in \{1, 2, \dots, k\}$

5. ปัญหาการจัดลำดับการผลิตที่มีหลายวัตถุประสงค์

ลักษณะสายการประกอบในงานวิจัยมีลักษณะดังนี้

- 1) เป็นสายการประกอบผลิตภัณฑ์ผสมแบบสองด้านที่มีความสมดุลที่อัตราส่วนเท่ากับสัดส่วนผลิตภัณฑ์
- 2) ผลิตภัณฑ์ที่ทำการผลิตจะมีหลากหลายรุ่น แต่มีคุณสมบัติคล้ายกันจึงสามารถผลิตบนสายการประกอบ

เดียวกันได้ และบางผลิตภัณฑ์จะไม่มีการทำงานบางชิ้นงาน (Task)

3) ในการจัดจะพิจารณาความต้องการผลิตภัณฑ์จาก Minimum Part Set (MPS)

4) สถานีนงานที่ใช้จัดเป็นแบบปิด ทราบจำนวนสถานีนงานที่อยู่ในสายการผลิตและเวลาดำเนินงานของชิ้นงานที่อยู่ในสถานีนงาน

5) ไม่นำระยะทางและเวลาการเดินทางของพนักงานเข้ามาพิจารณา

6) รอบเวลาในการปล่อยสินค้าเข้าสู่สายพานการผลิตมีค่าคงที่

7) การปรับตั้งเครื่องขึ้นอยู่กับลำดับงานที่อยู่ก่อนหน้า

5.1 ฟังก์ชันวัตถุประสงค์

ในงานวิจัยนี้พิจารณา 3 ฟังก์ชันวัตถุประสงค์ดังนี้ [2]

5.1.1 ความแปรผันในการผลิตที่น้อยที่สุด

สายการผลิตแบบผลิตภัณฑ์ผสมจะต้องป้อนงานเข้าสายการผลิตอย่างสม่ำเสมอและต่อเนื่อง ถ้าผลิตภัณฑ์มีความแตกต่างกันมากจะทำให้มีความแปรผันในการผลิตมาก ในการจัดลำดับการผลิตจึงมีความต้องการให้มีความแปรผันในการผลิตน้อยที่สุด ซึ่งสามารถคำนวณได้จากสมการที่ 6

$$f_1(x) = \sum_{i=1}^I \sum_{m=1}^M \left[x_{im} - i \frac{d_m}{I} \right]^2 \quad (6)$$

โดย

$f_1(x)$ คือความผันแปรในการผลิตในลำดับการผลิต
 M คือจำนวนชนิดของผลิตภัณฑ์ที่ถูกผลิตทั้งหมด
 D_m คือความต้องการของผลิตภัณฑ์ m โดย $m = 1, 2, \dots, M$

I คือจำนวนหน่วยทั้งหมดของทุกผลิตภัณฑ์ หรือความต้องการในการผลิตทั้งหมดในทุกผลิตภัณฑ์ โดย $I = \sum_{m=1}^M d_m$

$x_{i,m}$ คือจำนวนผลิตภัณฑ์ที่ผลิตในลำดับที่ i โดย $x_{i,m} = 1$ เมื่อผลิตภัณฑ์ในลำดับที่ i เป็นผลิตภัณฑ์ชนิด m และ $x_{i,m} = 0$ เมื่อผลิตภัณฑ์

ในลำดับที่ i ไม่ใช่ผลิตภัณฑ์ชนิด m

5.1.2 ปริมาณงานที่ไม่เสร็จที่น้อยที่สุด

ปริมาณงานที่ไม่เสร็จเกิดจากพนักงานในสถานีนงานได้รับภาระโหลดเกินจึงทำให้ไม่สามารถทำงานให้เสร็จตามเวลาที่กำหนดได้ จึงต้องมีพนักงาน (Utility Worker) เข้ามาช่วย การทำให้ปริมาณงานที่ไม่เสร็จน้อยที่สุดจะช่วยลดค่าใช้จ่ายพนักงานส่วนนี้ได้และช่วยลดความเสี่ยงในการหยุดของสายการผลิตลงได้ ซึ่งสามารถคำนวณได้จากสมการที่ 7

$$f_2(x) = \sum_{n_m=1}^{N_M} (\sum_{i=1}^I U_{i,n_m} + Z_{(i+1),n_m} / v_c) \quad (7)$$

เมื่อ

$$U_{i,n_m} = \left\{ \begin{array}{l} \max \left[0, \frac{(Z_{i,n_m} + v_c \sum_{m=1}^M X_{i,m} (t_{2n_m-1,m} + Y_{2n_m-1,m}) - L_{n_m})}{v_c} \right] \\ + \max \left[0, \frac{(Z_{i,n_m} + v_c \sum_{m=1}^M X_{i,m} (t_{2n_m,m} + Y_{2n_m,m}) - L_{n_m})}{v_c} \right] \end{array} \right\} \quad (8)$$

$Z_{(i+1),n_m} =$

$$\max \left\{ \begin{array}{l} \max[0, \min(Z_{i,n_m} + v_c \sum_{m=1}^M X_{i,m} (t_{2n_m-1,m} + Y_{2n_m-1,m}) - \gamma v_c L_{n_m} - \gamma v_c)] \\ \max[0, \min(Z_{i,n_m} + v_c \sum_{m=1}^M X_{i,m} (t_{2n_m,m} + Y_{2n_m,m}) - \gamma v_c L_{n_m} - \gamma v_c)] \end{array} \right\} \quad (9)$$

$f_2(x)$ คือปริมาณงานที่ไม่เสร็จในการผลิต

N_M คือคู่สถานีนงาน (Mated Station) โดย $n_m = 1, 2, \dots, N_M$

n_w คือคู่สถานีนงานด้านซ้ายและด้านขวา $n_w = 2n_m - 1$ และ $2n_m$ โดย $n_w = 1, 2, \dots, N_w = 2N_M$
 คือเวลาดำเนินการทั้งหมดของการผลิตหนึ่งรอบเวลาการผลิต หรือหนึ่งรอบ Model Sequence โดย

$$T = \sum_{n_m=1}^{N_M} \max \left[\sum_{m=1}^M (t_{2n_m-1,m} d_m + Y_{2n_m-1,m}), \sum_{m=1}^M (t_{2n_m,m} d_m + Y_{2n_m,m}) \right]$$

$t_{j,m}$ คือเวลาดำเนินการ (Processing Time) ทั้งหมดของผลิตภัณฑ์ m ที่สถานีนงาน j โดย $t_{j,m} = \sum_{k \in s_j} p_{km} + t_m$

p_{km} คือเวลาดำเนินงาน (Processing Time) ในชิ้นงาน k ของผลิตภัณฑ์ m

t_m คือเวลาปรับตั้งค่าเครื่องจักรของผลิตภัณฑ์ m

J คือสถานีนงาน

$Y_{j,m}$	คือเวลาเดินเปล่าทั้งหมด เนื่องมาจากลำดับก่อนหน้าของชั้นงานของผลิตภัณฑ์ m ที่สถานีงาน j
γ	คือช่วงเวลาการปล่อย (Launch Interval) ผลิตภัณฑ์เข้าสู่สายพานการผลิต โดย $\gamma = T/(I * N_M)$
w	คือระยะทางการเคลื่อนที่ย้อนกลับของพนักงานเพื่อเริ่มต้นผลิตผลิตภัณฑ์ถัดไป โดยที่ $w = \gamma \times v_c$
v_c	คือความเร็วของสายพานการผลิต
$x_{i,m}$	คือจำนวนหน่วยทั้งหมดของผลิตภัณฑ์ m ที่ถูกผลิตในลำดับการผลิต i เมื่อ $i = 1, 2, \dots, I$
U_{i,n_m}	คือปริมาณของ Utility Work (งานที่ไม่เสร็จ) ของผลิตภัณฑ์ลำดับที่ i ในคู่สถานีงาน n_m
$Z_{(i+1),n_m}$	คือเวลาเริ่มงานของผลิตภัณฑ์ลำดับที่ i ในคู่สถานีงาน n_m
L_j	คือความยาวของสถานีงาน j

5.1.3 เวลาการปรับตั้งเครื่องจักรที่น้อยที่สุด

ในการจัดลำดับการผลิตเวลาในการปรับตั้งเครื่องจักรจะขึ้นอยู่กับชนิดของผลิตภัณฑ์ก่อนหน้า การทำให้เวลาการปรับตั้งเครื่องจักรน้อยที่สุดจะทำให้สามารถลดเวลาในการผลิตลง ผลผลิตที่ได้จะเพิ่มขึ้น ซึ่งสามารถคำนวณได้จากสมการที่ 10

$$f_3(x) = \sum_{i=1}^I \sum_{m=1}^M s_i t_m \quad (10)$$

โดย

$f_3(x)$ คือเวลาของการปรับตั้งเครื่อง

s_i คือการปรับตั้งเครื่อง ถ้า $s_i = 1$ เมื่อมีการปรับตั้งเครื่องเกิดขึ้น และ $s_i = 0$ เมื่อไม่มีการปรับตั้งเครื่องเกิดขึ้น

5.2 จำนวนลำดับการผลิตทั้งหมด

จำนวนลำดับการผลิตทั้งหมดสามารถคำนวณได้จากสมการที่ 11 [1], [4]

$$\text{จำนวนลำดับการผลิตทั้งหมด} = \frac{(\sum_{m=1}^M d_m)!}{\prod_{m=1}^M (d_m)!} \quad (11)$$

โดย

M คือจำนวนของชนิดผลิตภัณฑ์

m คือชนิดของผลิตภัณฑ์

d_m คือจำนวนความต้องการของผลิตภัณฑ์ m

เมื่อขนาดของปัญหาเพิ่มขึ้นจำนวนของคำตอบที่เป็นไปได้จะเพิ่มขึ้นแบบ Exponential ปัญหาการจัดลำดับการผลิตจึงเป็นปัญหาแบบ NP-hard

6. อัลกอริทึม BBO

Simon [6] ได้มีการศึกษาและพัฒนาอัลกอริทึมใหม่ที่มีลักษณะเป็น Evolution Algorithm มีชื่อว่าการหาค่าเหมาะสมแบบการกระจายของสิ่งมีชีวิตตามภูมิศาสตร์ (Biogeography Based Optimization: BBO) เป็นการนำแนวคิดการเคลื่อนย้ายประชากรจากแหล่งหนึ่งไปอีกแหล่งหนึ่ง BBO จะมีค่าที่สำคัญ 4 ค่า คือ Habitat Suitability Index (HSI) เป็นค่า Fitness ที่ใช้วัดว่าคำตอบของประชากรเป็นคำตอบที่ดีหรือคำตอบที่แย่ ค่าที่สองคือ Suitability Index Variable (SIV) เป็นค่าลักษณะ (Feature) ของคำตอบแต่ละตัวของประชากรในเกาะ ค่าที่สามคือ Immigration Rate เป็นอัตราการอพยพเข้าของประชากรในเกาะ และค่าที่สี่คือ Emigration Rate เป็นอัตราการอพยพออกของประชากรในเกาะ ถ้าคำตอบของประชากรในเกาะมีค่าที่ดี ค่าของการอพยพเข้าจะมีค่าน้อย ค่าของการอพยพออกจะมีค่ามาก เนื่องจากประชากรของเกาะที่มีคำตอบที่ดีจะพยายามแบ่งลักษณะ (Feature) ของคำตอบไปให้ประชากรของเกาะที่มีคำตอบที่แย่กว่าเพื่อให้คำตอบในเกาะที่แย่นั้นดีขึ้น

ขั้นตอนของการทำงาน BBO ที่ใช้ในงานวิจัยมีดังนี้
ขั้นที่ 1 สร้างสตริงคำตอบเริ่มต้นด้วยวิธีการสุ่ม N ตัว จะใช้วิธีการสุ่มด้วยความน่าจะเป็นที่เท่ากัน [17] ปัญหาตัวอย่างได้กำหนดค่า MPS = 1:2:3 แสดงว่าในการจัดลำดับการผลิตนี้จะมีผลิตภัณฑ์ A, B และ C จำนวน 1, 2 และ 3 ชิ้น ตามลำดับ ความยาวของลำดับการผลิตเท่ากับ 6 จากนั้นทำการใส่รหัสงานให้กับผลิตภัณฑ์ที่ทำการจัดลำดับการผลิตดังตารางที่ 1 แล้วทำการสุ่ม

สตริงคำตอบ 5 สตริงคำตอบดังตารางที่ 2 แล้วทำการ
คัดลอกสตริงคำตอบไว้ในสตริงคำตอบชั่วคราว

ตารางที่ 1 รหัสงานผลิตภัณฑ์

Model Sequence	A	B	B	C	C	C
String	1	2	3	4	5	6

ตารางที่ 2 สตริงคำตอบเริ่มต้นที่ได้จากการสุ่ม

Strings	Model Sequences					
1	3	5	1	2	6	4
2	2	6	3	1	4	5
3	1	2	5	3	4	6
4	3	6	1	5	4	2
5	4	1	5	2	6	3

ขั้นที่ 2 ประเมินค่าฟังก์ชันวัตถุประสงค์ของสตริง
คำตอบ นำสตริงคำตอบเริ่มต้นที่สุ่มได้มาคำนวณค่า
ฟังก์ชันวัตถุประสงค์ 3 ฟังก์ชันวัตถุประสงค์ดังตารางที่ 3

ตารางที่ 3 ค่าฟังก์ชันวัตถุประสงค์และค่าความแข็งแรง

Strings	Variance of Production Rates	Utility Work	Setup Time
1	31.4556	15.6622	5.6877
2	40.1222	15.5409	5.6253
3	47.4556	16.9536	5.5354
4	22.6556	15.5363	5.6327
5	34.7889	15.8744	6.3763

ขั้นที่ 3 กำหนดค่าความแข็งแรงของสตริงคำตอบ
ด้วยวิธี Non-Dominated Sorting [18] ได้ผลดังตารางที่ 4
และกำหนดค่าความหนาแน่นด้วยวิธี Crowding Distance
[18] จะพิจารณาจากค่าความแข็งแรงที่เท่ากัน (ค่า Fitness
เท่ากัน) โดยเริ่มจากสตริงที่มีความแข็งแรงมากที่สุด
(ค่า Fitness=1) จนถึงความแข็งแรงน้อยที่สุดตามลำดับ
ถ้าสตริงไหนมีค่าความแข็งแรงค่าเดียวจะกำหนดค่า
Crowding Distance เท่ากับอนันต์ (Infinity) ดังตารางที่ 5

ตารางที่ 4 ค่าฟังก์ชันวัตถุประสงค์และค่าความแข็งแรง

Strings	Variance of Production Rates	Utility Work	Setup Time	Fitness
1	31.4556	15.6622	5.6877	2
2	40.1222	15.5409	5.6253	1
3	47.4556	16.9536	5.5354	1
4	22.6556	15.5363	5.6327	1
5	34.7889	15.8744	6.3763	3

ตารางที่ 5 ค่าความหนาแน่นของสตริงคำตอบ

Strings	Variance of Production Rates	Utility Work	Setup Time	Fitness	Crowding Distance
1	31.4556	15.6622	5.6877	2	Infinity
2	40.1222	15.5409	5.6253	1	3
3	47.4556	16.9536	5.5354	1	Infinity
4	22.6556	15.5363	5.6327	1	Infinity
5	34.7889	15.8744	6.3763	3	Infinity

ขั้นที่ 4 กำหนดค่าสปีชีส์เคาท์ การหาอัตราการอพยพ
(Migration Rate) และค่าความน่าจะเป็นของการเกิด
สปีชีส์เคาท์ โดยการกำหนดค่าสปีชีส์เคาท์จะพิจารณาค่า
ความแข็งแรงที่มากที่สุด (เท่ากับ 1) ค่าสปีชีส์เคาท์จะมีค่า
สูงที่สุดแล้วเรียงลำดับไปจนถึงค่าความแข็งแรงน้อยที่สุด
ซึ่งสามารถจัดลำดับได้ดังตารางที่ 6

ตารางที่ 6 ค่าสปีชีส์เคาท์ของสตริงคำตอบ

Strings	Variance of Production Rates	Utility Work	Setup Time	Fitness	Species Count
1	31.4556	15.6622	5.6877	2	2
2	40.1222	15.5409	5.6253	1	3
3	47.4556	16.9536	5.5354	1	3
4	22.6556	15.5363	5.6327	1	3
5	34.7889	15.8744	6.3763	3	1

การอพยพเข้า (Immigration Rate: λ) และการ
อพยพออก (Emigration Rate: μ) เป็นวิธีการที่สตริง
คำตอบพยายามที่จะแบ่งลักษณะ (Feature) ของคำตอบ
ไปให้สตริงคำตอบอื่นๆ ซึ่งค่า Immigration Rate (λ_k)
มีสมการดังนี้

$$\lambda_k = I \left(1 - \frac{k}{n} \right); n = K + 1 \quad (12)$$

ค่า Emigration Rate มีสมการดังนี้

$$\mu_k = E \left(\frac{k}{n} \right); n = K + 1 \quad (13)$$

ความน่าจะเป็นของการเกิดสปีชีส์เคาท์ k ($P_{K,k}$) มีสมการดังนี้

$$P_{K,k} = \frac{\left(\frac{I}{E}\right)^k \left(\frac{n!}{k!(n-k)!}\right)}{1 + \sum_{i=1}^n \left(\frac{I}{E}\right)^i \left(\frac{n!}{i!(n-i)!}\right)} \quad (14)$$

โดยกำหนดให้ค่า I คืออัตราการอพยพเข้าสูงสุด และ E อัตราการอพยพออกสูงสุดมีค่าเท่ากับ 1 และ $n = K + 1$ ได้ผลการคำนวณดังตารางที่ 7

ตารางที่ 7 การคำนวณค่าอพยพของสตริงคำตอบ

No.	String	Species Count	λ_k	μ_k	P_λ	P_μ	$P_{K,k}$
1	4	3	0.2500	0.7500	0.1667	0.5000	0.2500
2	2						
3	3						
4	1	2	0.5000	0.5000	0.3333	0.3333	0.3750
5	5	1	0.7500	0.2500	0.5000	0.1667	0.2500

ขั้นที่ 5 การอพยพและการมิวเตชัน หลังจากทำการคำนวณค่าอัตราการอพยพแล้ว จะเริ่มทำการอพยพสตริงคำตอบเพื่อแลกเปลี่ยนลักษณะที่ดีภายในสตริงคำตอบแล้วนำไปทำการมิวเตชันเพื่อให้เกิดความหลากหลายของสตริงคำตอบ

การอพยพ

การอพยพของสตริงคำตอบจะพิจารณาดังนี้

1) ทำการคัดลอกสตริงคำตอบเริ่มต้น (Y) ลงไปในสตริงคำตอบชั่วคราว (Z) เช่น

$Y_1 = 3 \ 5 \ 1 \ 2 \ 6 \ 4$

$Z_2 = 3 \ 5 \ 1 \ 2 \ 6 \ 4$

2) ทำการอพยพกับทุกสตริงคำตอบ โดยเริ่มจากสตริงคำตอบที่มีค่า Species Count สูงที่สุดไปจนถึงต่ำสุดในตารางมีสตริงคำตอบที่มีค่า Species Count สูงที่สุด 3 สตริงคำตอบ คือสตริงคำตอบที่ 2, 3 และ 4 จากนั้นทำการสุ่มด้วยค่าความน่าจะเป็นที่เท่ากัน เช่น ทำการ

ตารางที่ 9 การปรับปรุงคำตอบ

String	Bit	Immigration $Z_{k,n_k}(b)$	Emigration $X_{k,m_k}(b)$	การปรับปรุง	สตริงคำตอบ					
4	3	1	1	เริ่มต้น	3	6	<u>1</u>	5	4	2
				ปรับปรุง	3	6	<u>1</u>	5	4	2
				ซ่อมแซม	3	6	<u>1</u>	5	4	2
3	2	1	5	เริ่มต้น	1	2	5	3	4	6
				ปรับปรุง	1	<u>5</u>	5	3	4	6
				ซ่อมแซม	1	5	<u>2</u>	3	4	6

สุ่มออกมาได้สตริงที่ 4 คือ $Z_4 = 3 \ 6 \ 1 \ 5 \ 4 \ 2$

3) สร้างเลขสุ่ม (r) เพื่อทำการอพยพสตริงคำตอบ ถ้าค่าสุ่ม $r < P_{\lambda,k}$ จะทำการอพยพเข้า โดยจะใช้วงล้อรูเล็ตในการเลือกค่าที่จะอพยพออก โดยการสุ่มค่าของ P_μ ถ้าสุ่มได้ค่า Species Count ที่มีหลายสตริงคำตอบ ให้สุ่มเลือกสตริงคำตอบที่อยู่ใน Species Count มา 1 สตริงตัวอย่างเช่น เราจะเริ่มพิจารณาการอพยพที่ละบิตของสตริงที่ 4 คือ

$Z_{4(1)} = 3$ ค่า $P_{\lambda,3} = 0.1667$ โดยการสุ่มค่า r ระหว่าง 0 ถึง 1 ออกมา ค่าที่สุ่มออกมาได้ $r = 0.3155$ ซึ่งค่า $r > P_{\lambda,3}$ จะไม่ทำการอพยพ คงค่าไว้เหมือนเดิม

$Z_{4(2)} = 6$ ค่า $P_{\lambda,3} = 0.1667$ ค่าที่สุ่มออกมาได้ $r = 0.3007$ ซึ่งค่า $r > P_{\lambda,3}$ จะไม่ทำการอพยพ คงค่าไว้เหมือนเดิม

$Z_{4(3)} = 1$ ค่า $P_{\lambda,3} = 0.1667$ ค่าที่สุ่มออกมาได้ $r = 0.0420$ ซึ่งค่า $r < P_{\lambda,3}$ แสดงว่ามีการอพยพเข้า เราจึงใช้วงล้อรูเล็ตเลือกค่าที่จะอพยพออก โดยการสุ่มค่าของ P_μ ซึ่งเราสุ่มได้ $P_\mu = 0.5279$ ซึ่งจะไปตกอยู่ที่สปีชีส์เคาท์ที่ 2 ซึ่งมีสตริงคำตอบที่ 1 เพียงสตริงคำตอบเดียว จึงนำสตริงคำตอบที่ 1 ตรงตำแหน่งที่ $Y_{1(3)} = 1$ ทำการอพยพเข้าไปใน $Z_{3(2)}$ ดังตารางที่ 8

ตารางที่ 8 การอพยพของสตริงคำตอบ

Immigrate	Emigrate			
bit for $r < P_{\lambda,k}$	select species count from P_μ	string in specie count k	select string	$x_{k,m_k}(b)$
3	2	{1}	{1}	1
2	2	{1}	{1}	5

4) ค่าสตริงคำตอบที่ได้มีตัวเลขที่ซ้ำกัน จึงทำการซ่อมแซมคำตอบโดยการนำสตริงคำตอบ Y_4 มาแทนจะได้จะได้สตริงคำตอบ $Y_4=3\ 6\ 1\ 5\ 4\ 2$ ดังตารางที่ 9

การมิวเตชัน

เมื่อทำการอพยพสตริงคำตอบเรียบร้อยแล้วจะเข้าสู่กระบวนการมิวเตชันโดยนำสตริงคำตอบ Z มาทำการสลับตำแหน่งภายในสตริงคำตอบโดยใช้วิธี Reciprocal Exchange Mutation [17] ในการมิวเตชันสตริงคำตอบซึ่งมีขั้นตอนดังนี้

1) เลือกสปีชีส์เคาท์ที่จะทำมิวเตชันโดยการใช้วงล้อรูเล็ตทำการสุ่มค่า r ที่อยู่ระหว่าง 0-1 ให้กับสปีชีส์เคาท์ “ถ้าค่า $q_{k-1} \leq r < q_k; q_0 = 0$ สปีชีส์เคาท์จะถูกเลือกไปทำการมิวเตชัน ซึ่งค่าความน่าจะเป็นในการเลือกสามารถคำนวณได้จากสมการที่ 15

$$m(k) = \frac{1-P_{K,k}}{\sum_{k=1}^K (1-P_{K,k})} \quad (15)$$

โดยที่ $m(k)$ คือ ความน่าจะเป็นในการเลือกสปีชีส์เคาท์ k $P_{K,k}$ คือ ความน่าจะเป็นของการเกิดสปีชีส์เคาท์ จากตัวอย่างทำการสุ่มค่าได้ $r = 0.3355$ ซึ่งจะปตกอยู่ที่สปีชีส์เคาท์ 3 (ตารางที่ 10) ซึ่งมีสตริงคำตอบ 3 สตริงคำตอบ คือสตริงคำตอบที่ 4, 2 และ 3

ตารางที่ 10 ความน่าจะเป็นในการมิวเตชัน

No.	String	Species Count	$P_{K,k}$	$1 - P_{K,k}$	$m(k)$	q_k	r
1	4	3	0.2500	0.7500	0.3529	0.3529	Selected
2	2						
3	3	2	0.3750	0.6250	0.2941	0.6470	-
4	1						
5	5	1	0.2500	0.7500	0.3529	1	-

2) เมื่อได้สปีชีส์เคาท์ที่จะทำการมิวเตชันแล้ว ให้นำสตริงคำตอบที่อยู่ในสปีชีส์เคาท์ทำการสลับบิตกันโดยใช้ค่าความน่าจะเป็นในการมิวเตชัน (P_m)=0.01 [19] ทำการคูณกับจำนวนบิตทั้งหมดในสตริงคำตอบ $0.01 \times 18 = 0.18$ บิต เพราะฉะนั้นจะทำการสลับบิต โดยสุ่มตำแหน่งภายในสตริงคำตอบ 2 ตำแหน่งมาทำการสลับตำแหน่งกัน

จากตัวอย่างสตริงคำตอบที่ 4 ทำการสุ่มได้ตำแหน่งที่ 3 สลับกับตำแหน่งที่ 1 (ตารางที่ 11)

ตารางที่ 11 สตริงคำตอบที่ทำการมิวเตชัน

String ก่อนมิวเตชัน	3	6	1	5	4	2
String หลังมิวเตชัน	3	6	5	1	4	2

ขั้นที่ 6 การเก็บสตริงคำตอบที่ดีที่สุดและสตริงคำตอบที่ใช้ในรอบถัดไป หลังจากทำการมิวเตชันเสร็จจะได้สตริงคำตอบรุ่นลูก เพื่อนำไปรวมกับสตริงคำตอบเริ่มต้น และทำการเก็บค่าที่ดีที่สุดไว้

ขั้นที่ 7 การเก็บสตริงคำตอบที่ดีที่สุดและสตริงคำตอบที่ใช้ในรอบถัดไป การใช้เทคนิคการเก็บค่าที่ดีที่สุดเพื่อป้องกันไม่ให้คำตอบที่ดีหลังจากผ่านกระบวนการปรับปรุงต่างๆ เช่น การอพยพ และการมิวเตชันสูญหายไป เพราะในกระบวนการปรับปรุงอาจทำให้เกิดคำตอบที่แย่กว่าคำตอบก่อนปรับปรุงได้ จึงทำการเก็บค่าสตริงคำตอบเริ่มต้น (Y) และสตริงคำตอบคำตอบชั่วคราว (Z) มารวมกันแล้วหาค่าที่ดีที่สุดจากนั้นทำการเก็บค่าที่ดีที่สุดไว้เท่ากับจำนวนของสตริงคำตอบเริ่มต้น

ขั้นที่ 8 ทำการวนซ้ำตั้งแต่ขั้นตอนที่ 6.2 จนกระทั่งครบตามจำนวนรอบการทำงานที่กำหนด

ขั้นที่ 9 หยุดกระบวนการเมื่อครบจำนวนรอบการทำงานที่กำหนดไว้

7. การทดลอง

ในการทดลองจะใช้ปัญหามาตรฐานในการจัดลำดับการผลิตของ McMullen [20] มีรายละเอียดดังแสดงในตารางที่ 12 ทั้งหมด 12 ปัญหา โดยจะแบ่งเป็นปัญหามาตรกลาง และปัญหามาตรใหญ่ มีความแตกต่างกันในเรื่องของจำนวน 65, 148 และ 205 ชิ้นงาน มีจำนวนผลิตภัณฑ์ 4, 5, 10 และ 15 ผลิตภัณฑ์ และมีความแตกต่างของความต้องการสัดส่วนผลิตภัณฑ์ที่แตกต่างกัน

7.1 อัลกอริทึม

ในงานวิจัยนี้มีการนำอัลกอริทึม NSGAI และ DPSO มาเปรียบเทียบกับอัลกอริทึม BBO อัลกอริทึม NSGAI

[18] เป็นวิธีการถ่ายทอดลักษณะทางพันธุกรรมจากพ่อแม่ไปสู่รุ่นลูกหลาน โดยการคัดเลือกสตรึงคำตอบรุ่นพ่อแม่ที่เป็นสตรึงคำตอบที่ดี จากนั้นทำการครอสโอเวอร์และทำมิวเตชันเพื่อให้ได้สตรึงคำตอบที่ดีขึ้น แล้วนำไปเป็นสตรึงคำตอบรุ่นพ่อแม่ในเจนเนอเรชันถัดไป ส่วนอัลกอริทึม DPSO [21] เป็นวิธีการทางฮิวริสติกที่นำวิธีการค้นหาคำตอบโดยการเลียนแบบพฤติกรรมบินหาอาหารของฝูงนกจากการจดจำเส้นทางจากประสบการณ์ที่ผ่านมาของตัวเองและของสมาชิกในฝูง เคลื่อนที่ไปในจุดหมายและทิศทางต่างๆ แล้วทำการจดจำคำตอบที่ได้พบ แล้วนำคำตอบที่ได้ทำการปรับปรุงค่าทิศทางการเคลื่อนที่และตำแหน่งเพื่อให้ได้สตรึงคำตอบที่ใกล้เคียง

กับคำตอบที่ดีที่สุด

7.2 ตัววัดสมรรถนะ

ในการเปรียบเทียบสมรรถนะของอัลกอริทึม BBO NSGAI และ DPSO โดยจะใช้ตัวแปรตอบสนอง 4 ตัวในการวิเคราะห์ คือ การลู่เข้าสู่กลุ่มคำตอบที่แท้จริง (Convergence to the Pareto Optimal Set) [22] การกระจายตัวของกลุ่มคำตอบที่ค้นพบ (Spread Measurement) [23] อัตราส่วนของจำนวนกลุ่มคำตอบที่ค้นพบเทียบกับกลุ่มคำตอบที่แท้จริง (Ratio of Non-dominate Solution) [23] และเวลาที่ใช้ในการคำนวณ (Computation Time to Solution)

ตารางที่ 12 ปัญหาที่ใช้ในการวิจัย

ปัญหาการทดลอง	จำนวนชนิด ผลิตภัณฑ์	สัดส่วนผลิตภัณฑ์ (MPS)	ความยาวสตรึง	จำนวนคำตอบ ที่เป็นไปได้
Set 1	1	5	5:3:2:1:1	332640
	2	5	4:4:2:1:1	415800
Set 2	3	5	7:3:2:2:1	10810800
	4	5	4:3:3:3:2	126126000
Set 3	5	5	8:7:2:2:1	2.993E+09
	6	5	5:4:4:4:3	2.444E+11
Set 4	7	10	7:5:1:1:1:1:1:1	4.023E+12
	8	10	4:4:4:2:1:1:1:1:1	2.112E+15
Set 5	9	15	20:20:20:15:15:1:1:1:1:1:1:1:1:1	3.79E+78
	10	15	15:15:10:10:10:10:10:10:4:1:1:1:1:1	9.959E+92
Bartholdi 148	4		7:3:4:6	4.655E+09
Lee 205	4		5:7:8:10	2.997E+15

ตารางที่ 13 ค่าพารามิเตอร์ของอัลกอริทึม

Parameter Settings	NSGAI		DPSO		BBO	
Population size	100		100		100	
Number of Swarm and Number of Particles in each swarm			P1.1:10-10	P1.2:5-20		
			P2.1:10-10	P2.2:10-10		
			P3.1:10-10	P3.2:10-10		
			P4.1:4-25	P4.2:4-25		
			P5.1:5-20	P5.2:5-20		
			Bartholdi:4-25	Lee:4-25		
Crossover method	Weight mapping crossover					
Mutation Method	Reciprocal exchange mutation				Reciprocal exchange mutation	
Crossover probability and Mutation Probability	P1.1:0.8-0.1	P1.2:0.8-0.2			0.01	
	P2.1:0.9-0.1	P2.2:0.9-0.2				
	P3.1:0.8-0.3	P3.2:0.7-0.3				
	P4.1:0.9-0.1	P4.2:0.6-0.4				
	P5.1:0.9-0.2	P5.2:0.7-0.4				
	Bartholdi: 0.8-0.3	Lee:0.6-0.1				
Learning factors			0.1			
Inertia weight :			1			
Migration model					P1.1:S	P1.2:S
					P2.1:S	P2.2:S
					P3.1:L	P3.2:L
					P4.1:L	P4.2:L
					P5.1:L	P5.2:S
					Bartholdi:L	Lee:S
Learning effect	80%		80%		80%	

ตารางที่ 14 ผลการทดลอง

Efficiency	Algorithm	Problems											
		Set 1.1	Set 1.2	Set 2.1	Set 2.2	Set 3.1	Set 3.2	Set 4.1	Set 4.2	Set 5.1	Set 5.2	Bartholdi	Lee
Convergence	NSGAI	0.0253	0.0288	0.0232	0.0403	0.0345	0.0468	0.0470	0.0660	0.1141	0.1007	0.0406	0.0513
	DPSO	0.0245	0.0461	0.0451	0.0682	0.0377	0.0887	0.1036	0.1327	0.1679	0.1423	0.0794	0.1407
	BBO	0.0243	0.0249	0.0171	0.0082	0.0035	0.0052	0.0057	0.0038	0.0270	0.0165	0.0135	0.0101
Spread	NSGAI	0.5736	0.6190	0.7124	0.6822	0.5892	0.6194	0.6771	0.6688	0.6007	0.6883	0.6566	0.7504
	DPSO	0.5606	0.6136	0.6210	0.6127	0.6623	0.5785	0.6009	0.4789	0.5327	0.5331	0.4984	0.5846
	BBO	0.5033	0.5832	0.6053	0.6385	0.5707	0.5742	0.7946	0.6141	0.5088	0.5536	0.5420	0.5960
Ratio	NSGAI	0.4095	0.4375	0.2424	0.1223	0.1062	0.0774	0.0927	0.1002	0.1140	0.1786	0.3762	0.1393
	DPSO	0.2000	0.1406	0.1341	0.0489	0.0531	0.0229	0.0083	0.0022	0.0439	0.0500	0.0198	0.0000
	BBO	0.5000	0.5469	0.6447	0.8287	0.8408	0.8997	0.8990	0.8976	0.8421	0.7714	0.6040	0.8607
Time(s)	NSGAI	304920	62847.5	663328	234640	38583.3	200793	992000	295525	93051.4	88903.5	24746.7	199528
	DPSO	8282.47	2584.65	15990.4	6793.67	22604.8	9491.06	5975.75	6816.25	12513.3	22183.8	5736	9193.21
	BBO	244750	47370.1	260559	132634	106253	91345.2	760865	146784	41945.2	46171.5	19213.9	98564.2

ตารางที่ 15 ผลการเปรียบเทียบการทดลองของแต่ละอัลกอริทึม

Efficiency	Algorithm	Problems													
		Set 1.1	Set 1.2	Set 2.1	Set 2.2	Set 3.1	Set 3.2	Set 4.1	Set 4.2	Set 5.1	Set 5.2	Bartholdi	Lee	%	Std.
Convergence	NSGAI	97.47	97.12	97.68	95.97	96.55	95.32	95.3	93.4	88.59	89.93	95.94	94.87	94.85	2.88
	DPSO	97.55	95.39	95.49	93.18	96.23	91.13	89.64	86.73	83.21	85.77	92.06	85.93	91.03	4.76
	BBO	97.57	97.51	98.29	99.18	99.65	99.48	99.43	99.62	97.3	98.35	98.65	98.99	98.67	0.86
Spread	NSGAI	42.64	38.1	28.76	31.78	41.08	38.06	32.29	33.12	39.93	31.17	34.34	24.96	34.69	5.34
	DPSO	43.94	38.64	37.9	38.73	33.77	42.15	39.91	52.11	46.73	46.69	50.16	41.54	42.69	5.41
	BBO	49.67	41.68	39.47	36.15	42.93	42.58	20.54	38.59	49.12	44.64	45.8	40.4	40.96	7.60
Ratio	NSGAI	40.95	43.75	24.24	12.23	10.62	7.74	9.27	10.02	11.4	17.86	37.62	13.93	19.97	13.35
	DPSO	20	14.06	13.41	4.89	5.31	2.29	0.83	0.22	4.39	5	1.98	0	6.03	6.38
	BBO	50	54.69	64.47	82.87	84.08	89.97	89.9	89.76	84.21	77.14	60.4	86.07	76.13	14.67

7.3 การกำหนดค่าพารามิเตอร์

เนื่องจากปัญหาที่ใช้ในการวิจัยที่แตกต่างกัน 12 ปัญหา จึงต้องมีการทำการทดสอบค่าพารามิเตอร์ที่เหมาะสมกับแต่ละปัญหาเพื่อให้ค่าของคำตอบที่ค้นหามีประสิทธิภาพมากที่สุด โดยทำการออกแบบการทดลองแบบ Full Factorial Design ทำซ้ำ 2 Replicate ปัญหาการทดลอง Set 1-4 จำนวนรอบการทำงาน 1500 รอบ และปัญหาการทดลอง Set 5, Bartholdi (148) และ Lee (205) จำนวนรอบการทำงาน 2000 รอบ ซึ่งมีรายละเอียดดังแสดงในตารางที่ 13 โดยการรันโปรแกรม Matlab 2009a บนคอมพิวเตอร์ Intel(R) Core™ i7-2670QM CPU@ 2.2GHz Ram 4 GB Window

7.4 ผลการทดลอง

จากผลการทดลองในตารางที่ 14 เมื่อพิจารณาโดยการนำมาเปรียบเทียบเป็นเปอร์เซ็นต์ดังตารางที่ 15 ผลการทดลองตัวชี้วัดสมรรถนะด้านการเข้าสู่กลุ่มคำตอบที่แท้จริง พบว่าทุกปัญหา BBO นั้นมีค่าการเข้าสู่กลุ่มคำตอบที่แท้จริงเข้าใกล้ 0 มากที่สุด รองลงมาคือ NSGAI และ DPSO ส่วนการพิจารณาตัวชี้วัดสมรรถนะด้านการกระจายของกลุ่มคำตอบที่ค้นพบ พบว่าทุกอัลกอริทึมมีการกระจายของกลุ่มคำตอบที่ไม่แตกต่างกัน ในการพิจารณาด้านอัตราส่วนของจำนวนกลุ่มคำตอบที่ค้นพบเทียบกับกลุ่มคำตอบที่แท้จริง พบว่าวิธี BBO มีค่าอัตราส่วนเข้าใกล้ค่า 1 มากที่สุด รองลงมาคือ NSGAI และ DPSO

และเมื่อพิจารณาด้านเวลาในการค้นหาคำตอบ พบว่าวิธี DPSO ใช้เวลาในการค้นหาคำตอบน้อยที่สุด รองลงมาคือ BBO ส่วน NSGAII นั้นจะใช้เวลาในการค้นหาคำตอบมากที่สุด

8. สรุป

การแก้ปัญหาการจัดลำดับการผลิตที่มีหลายวัตถุประสงค์บนสายการประกอบผลิตภัณฑ์ผสมแบบสองด้านภายใต้ผลกระทบจากการเรียนรู้โดยการนำเอา อัลกอริทึม BBO เข้ามาประยุกต์ใช้ พบว่า ตัวชี้วัดสมรรถนะของคำตอบที่ได้จากอัลกอริทึม BBO มีประสิทธิภาพที่ดีกว่า NSGA-II และ DPSO ทั้งในด้านการรู้เข้าของกลุ่มคำตอบที่แท้จริงและอัตราส่วนของจำนวนกลุ่มคำตอบที่ค้นพบเทียบกับกลุ่มคำตอบที่แท้จริง รวมถึง BBO ใช้เวลาในการหาคำตอบน้อยกว่า NSGA-II มาก ซึ่งแสดงให้เห็นว่า BBO มีความเหมาะสมในการแก้ปัญหาในงานวิจัยนี้

เอกสารอ้างอิง

- [1] O. Akgunduz and S. Tunal, "A review of the current applications of genetic algorithms in mixed-model assembly line sequencing," *International Journal of Production Research*, pp. 1-21, 2010.
- [2] C.J. Hyun, Y. Kim, and Y.K. Kim, "A genetic algorithm for multiple objective sequencing problems in mixed model assembly lines," *Computers and Operations Research* 25, pp. 675-690, 1998.
- [3] P.R. McMullen, P. Tarasewich, and G.V. Frazier, "Using genetic algorithms to solve the multi-product JIT sequencing problem with set-ups," *International Journal of Production Research* 38, pp. 2653-2670, 2000.
- [4] S.A. Mansouri, "A multi-objective genetic algorithm for mixed-model sequencing on JIT assembly lines," *European Journal of Operational Research* 167, pp. 696-716, 2005.
- [5] A.R. Rahimi-Vahed, M. Rabbani, R. Tavakkoli-Moghaddam, S.A. Torabi, and F. Jolai, "A multi-objective scatter search for a mixed-model assembly line sequencing problem," *Advanced Engineering Informatics*, vol. 21, pp. 85-99, 2007.
- [6] D. Simon, "Biogeography-Based Optimization," *IEEE transactions on evolutionary computation* 12, pp. 702-713, 2008.
- [7] J. Abell and D. Du, "A Framework for Multi-objective, Biogeography Based Optimization of Complex System Families," in *AIAA/ISSMO Multidisciplinary Analysis Optimization Conference Fort worth Texas*, 2010.
- [8] H. Mo and L. Xu, "Biogeography Migration Algorithm for Traveling Salesman Problem," in *International Conference on Swarm Intelligence Beijing*, 2010.
- [9] D. Biskup, "Single-machine scheduling with learning considerations," *European Journal of Operational Research* 115, pp. 173-178, 1999.
- [10] P. Chutima and P. Chimklai, "Multi-objective two-sided mixed-model assembly line balancing using particle swarm optimisation with negative knowledge," *Computers & Industrial Engineering*, pp. 39-55, 2012.
- [11] U. Ozcan and B. Toklu, "Balancing of mixed-model two-sided assembly lines," *Computers & Industrial Engineering*, pp. 217-227, 2009.
- [12] Y.K. Kim, Y. H. Kim, and Y. J. Kim, "Two-sided assembly line balancing: a genetic algorithm approach," *Production Planning & Control* 11, pp. 44-53, 2000.
- [13] T.P. Wright, "Factors affecting the cost of

- airplanes,” *Journal of Aeronautical Sciences* 3, pp. 122-128, 1936.
- [14] D. Biskup, “A state-of-the-art review on scheduling with learning effects,” *European Journal of Operational Research* 188, pp. 315-329, 2008.
- [15] A. Konak, D. W. Coit, and E. S. Alice, “Multi-objective optimization using genetic algorithms: A tutorial,” *Reliability Engineering and System Safety* 91, pp. 992-1007, 2006.
- [16] N. Srinivas and K. Deb, “Multiobjective Optimization Using Nondominated Sorting in Genetic Algorithms,” *the Journal of Evolutionary Computation*, pp. 221-248, 2002.
- [17] Y.K. Kim, C. J. Hyun, and Y. Kim, “Sequencing in mixed model assembly lines: A genetic algorithm approach,” *Computers and Operations Research*, vol. 23, no.12, pp. 1131-1145, 1996.
- [18] K. Deb, A. Pratap, S. Agarwal, T. Meyarivan, “A fast and elitist multiobjective genetic algorithm: NSGA-II,” *IEEE Transactions on Evolutionary Computation*, pp. 182-197, 2002.
- [19] H. Ma, “An analysis of the equilibrium of migration models for biogeography-based optimization,” *Information Sciences* 180, pp. 3444-3464, 2010.
- [20] P.R. McMullen, “An efficient frontier approach to addressing JIT sequencing problems with setups via search heuristics,” *Computers and Industrial Engineering* 41, pp. 335-353, 2001.
- [21] C.J. Liao, C.T. Tseng, and P. Luarn, “A discrete version of particle swarm optimization for flowshop scheduling problems,” *Computers & Operations Research*, pp. 3099-3111, 2007.
- [22] R. Kumar, and P.K. Singh, “Pareto evolutionary algorithm hybridized with local search for bi-objective TSP,” *Studies in Computational Intelligence*, pp. 361-398, 2007.
- [23] M. Li, and J. Zheng, *Spread assessment for evolutionary multi-objective optimization*, Springer-Verlag Berlin Heidelberg, 2009, pp. 216-230.